



US008928730B2

(12) **United States Patent**
Stec et al.

(10) **Patent No.:** **US 8,928,730 B2**
(45) **Date of Patent:** **Jan. 6, 2015**

(54) **METHOD AND SYSTEM FOR CORRECTING A DISTORTED INPUT IMAGE**

(75) Inventors: **Piotr Stec**, Galway (IE); **Alexei Pososin**, Galway (IE); **Mihai Munteanu**, Brasov (RO); **Corneliu Zaharia**, Brasov (RO)

(73) Assignee: **DigitalOptics Corporation Europe Limited**, Ballybrit, Galway (IE)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 370 days.

(21) Appl. No.: **13/541,650**

(22) Filed: **Jul. 3, 2012**

(65) **Prior Publication Data**

US 2014/0009568 A1 Jan. 9, 2014

(51) **Int. Cl.**

G06T 5/00 (2006.01)
H04N 5/225 (2006.01)

(52) **U.S. Cl.**

USPC **348/36**; 382/276

(58) **Field of Classification Search**

USPC 348/36
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

1,906,509 A	5/1933	Aschenrenner	
3,251,283 A	5/1966	Wood	
3,356,002 A	12/1967	Raitiere	
4,555,168 A	11/1985	Meier et al.	
5,508,734 A	4/1996	Baker et al.	
5,850,470 A	12/1998	Kung et al.	
6,044,181 A *	3/2000	Szeliski et al.	382/284
6,222,683 B1	4/2001	Hoogland et al.	
6,466,254 B1	10/2002	Furlan et al.	
6,750,903 B1	6/2004	Miyatake et al.	

7,058,237 B2	6/2006	Liu et al.
7,280,289 B2	10/2007	Yamakawa
7,327,899 B2	2/2008	Liu et al.
7,495,845 B2	2/2009	Asami
7,499,638 B2	3/2009	Arai et al.
7,612,946 B2	11/2009	Kweon et al.
7,613,357 B2	11/2009	Owechko et al.
7,835,071 B2	11/2010	Izumi
7,843,652 B2	11/2010	Asami et al.
7,929,221 B2	4/2011	Ning
8,090,148 B2	1/2012	Asari et al.

(Continued)

FOREIGN PATENT DOCUMENTS

JP 11-298780 A 10/1999

OTHER PUBLICATIONS

Non-Final Rejection, dated Dec. 20, 2012, for U.S. Appl. No. 13/234,146, filed Sep. 15, 2011.

(Continued)

Primary Examiner — Sath V Perungavoor

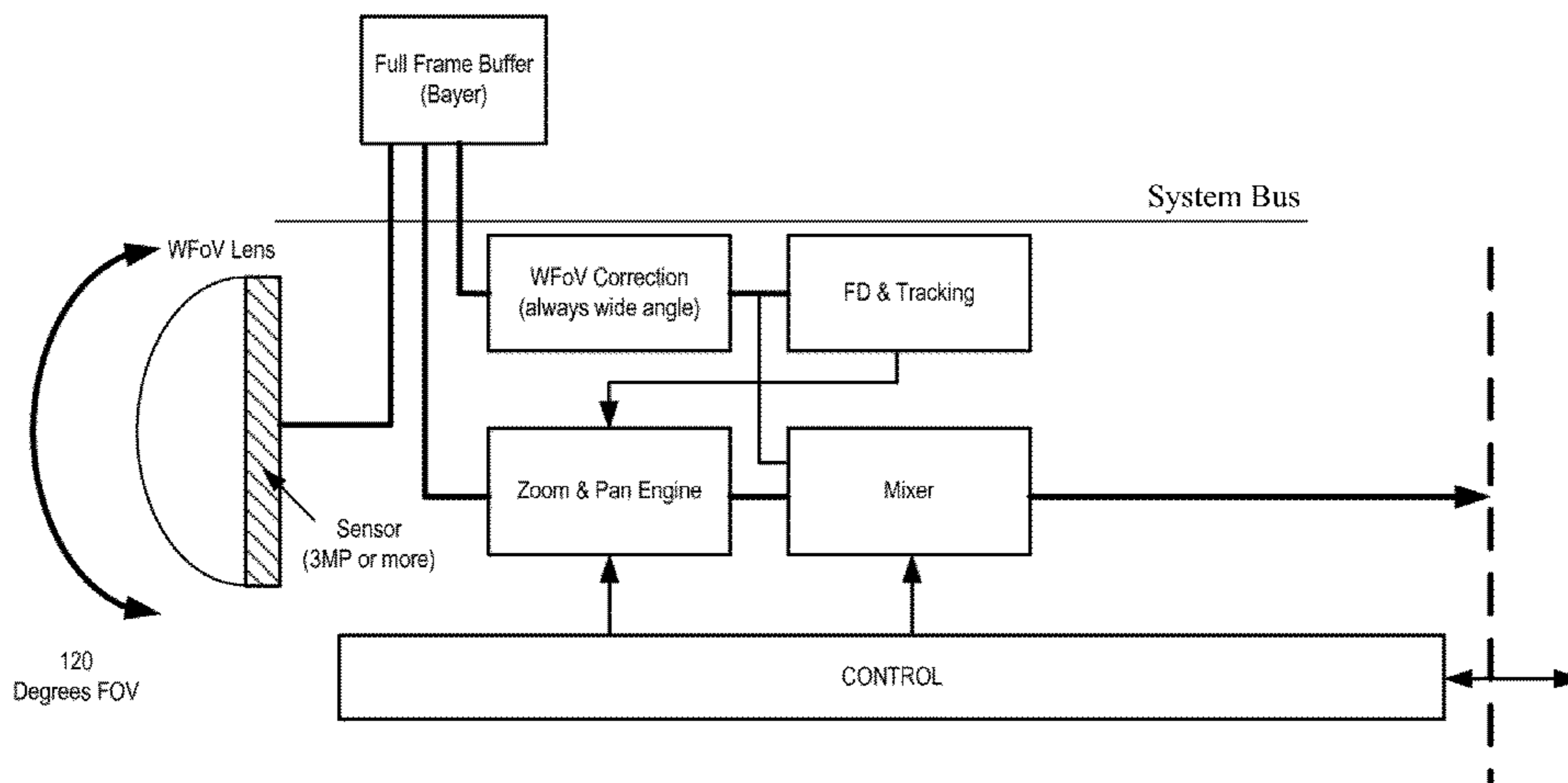
Assistant Examiner — Howard D Brown, Jr.

(74) *Attorney, Agent, or Firm* — Hickman Palermo Truong Becker Bingham Wong LLP

(57) **ABSTRACT**

A method for correcting a distorted input image includes determining a local region of an image to be displayed and dividing the region into an array of rectangular tiles, each tile corresponding to a distorted tile with a non-rectangular boundary within the input image. For each tile of the local region, maximum and minimum memory address locations of successive rows of the input image sufficient to span the boundary of the distorted tile are determined. Successive rows of the distorted input from between the maximum and minimum addresses are read. Distortion of the non-rectangular portion of the distorted input image is corrected to provide a tile of a corrected output image which is stored.

57 Claims, 9 Drawing Sheets



(56)

References Cited

U.S. PATENT DOCUMENTS

8,094,183	B2	1/2012	Toyoda et al.	
8,116,587	B2 *	2/2012	Shoaib et al.	382/276
8,134,479	B2	3/2012	Suhr et al.	
8,144,033	B2	3/2012	Chinomi et al.	
8,212,882	B2	7/2012	Florea et al.	
8,264,524	B1	9/2012	Davey	
8,311,344	B2	11/2012	Dunlop et al.	
8,340,453	B1 *	12/2012	Chen et al.	382/254
8,379,014	B2	2/2013	Wiedemann et al.	
2005/0084175	A1 *	4/2005	Olszak	382/284
2006/0140449	A1	6/2006	Otsuka et al.	
2007/0172150	A1	7/2007	Quan et al.	
2008/0075352	A1	3/2008	Shibuya et al.	
2010/0002071	A1 *	1/2010	Ahiska	348/36
2010/0033551	A1	2/2010	Agarwala et al.	
2010/0111440	A1	5/2010	Chai et al.	
2010/0283868	A1	11/2010	Clark et al.	
2011/0216156	A1 *	9/2011	Bigioi et al.	348/36
2012/0162454	A1 *	6/2012	Park et al.	348/208.6
2012/0249725	A1	10/2012	Corcoran et al.	
2012/0249727	A1	10/2012	Corcoran et al.	
2012/0249841	A1	10/2012	Corcoran et al.	
2013/0070126	A1 *	3/2013	Albu	348/241
2013/0114735	A1 *	5/2013	Wang	375/240.23
2013/0156279	A1 *	6/2013	Schoenmeyer et al.	382/128
2014/0055616	A1	2/2014	Corcoran et al.	
2014/0063229	A1 *	3/2014	Olsson et al.	348/84
2014/0152683	A1 *	6/2014	Nystad et al.	345/582

OTHER PUBLICATIONS

Non-Final Rejection, dated Mar. 15, 2013, for U.S. Appl. No. 12/959,089, filed Dec. 2, 2010.

Non-Final Rejection, dated Mar. 29, 2013, for U.S. Appl. No. 12/959,137, filed Dec. 2, 2010.

Non-Final Rejection, dated Mar. 15, 2013, for U.S. Appl. No. 12/959,151, filed Dec. 2, 2010.

Patent Abstracts of Japan, publication No. 11-298780, publication date: Oct. 29, 1999, for Wide-Area Image-Pickup Device and Spherical Cavity Projection Device.

Sang-Yong Lee, Jae-Tack Yoo, Yogendera Kumar, and Soo-Won Kim, Reduced Energy-Ratio Measure for Robust Autofocusing in Digital Camera, IEEE Signal Processing Letters, vol. 16, No. 2, Feb. 2009, pp. 133-136.

Jaehwan Jeon, Jinhee Lee, and Joonki Paik, Robust Focus Measure for Unsupervised Auto-Focusing Based on Optimum Discrete Cosine Transform Coefficients, IEEE Transactions on Consumer Electronics, vol. 57, No. 1, Feb. 2011, pp. 1-5.

Aaron Deever, In-Camera All-Digital Video Stabilization, ICIS '06 International Congress of Imaging Science, Final Program and Proceedings, Society for Imaging Science and Technology, pp. 190-193.

Felix Albu, Corneliu Florea, Adrian Zamfir, Alexandru Drimborean, 10.4-3 Low Complexity GlobalMotion Estimation Techniques for Image Stabilization, IEEE, Aug. 2008, pp. 1-4244-1459.

Masahiro Watanabe, Shree K. Nayar: Short Papers Telecentric Optics for Focus Analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19, No. 12, Dec. 1997, pp. 1360-1365.

U.S. Appl. No. 13/541,650, filed Jul. 3, 2012, entitled: Method and System for Correcting a Distorted Input Image.

A. Ben Hamza, Yun He, Hamid Krim, and Alan Willsky: A multiscale approach to pixel-level image fusion, Integrated Computer-Aided Engineering, IOS Press, No. 12, 2005 pp. 135-146 CIPA DC-007-Translation-2009: Multi-Picture Format, Established on Feb. 4, 2009, prepared by Standardization Committee, published by Camera & Imaging Products Association, 61 pages.

* cited by examiner

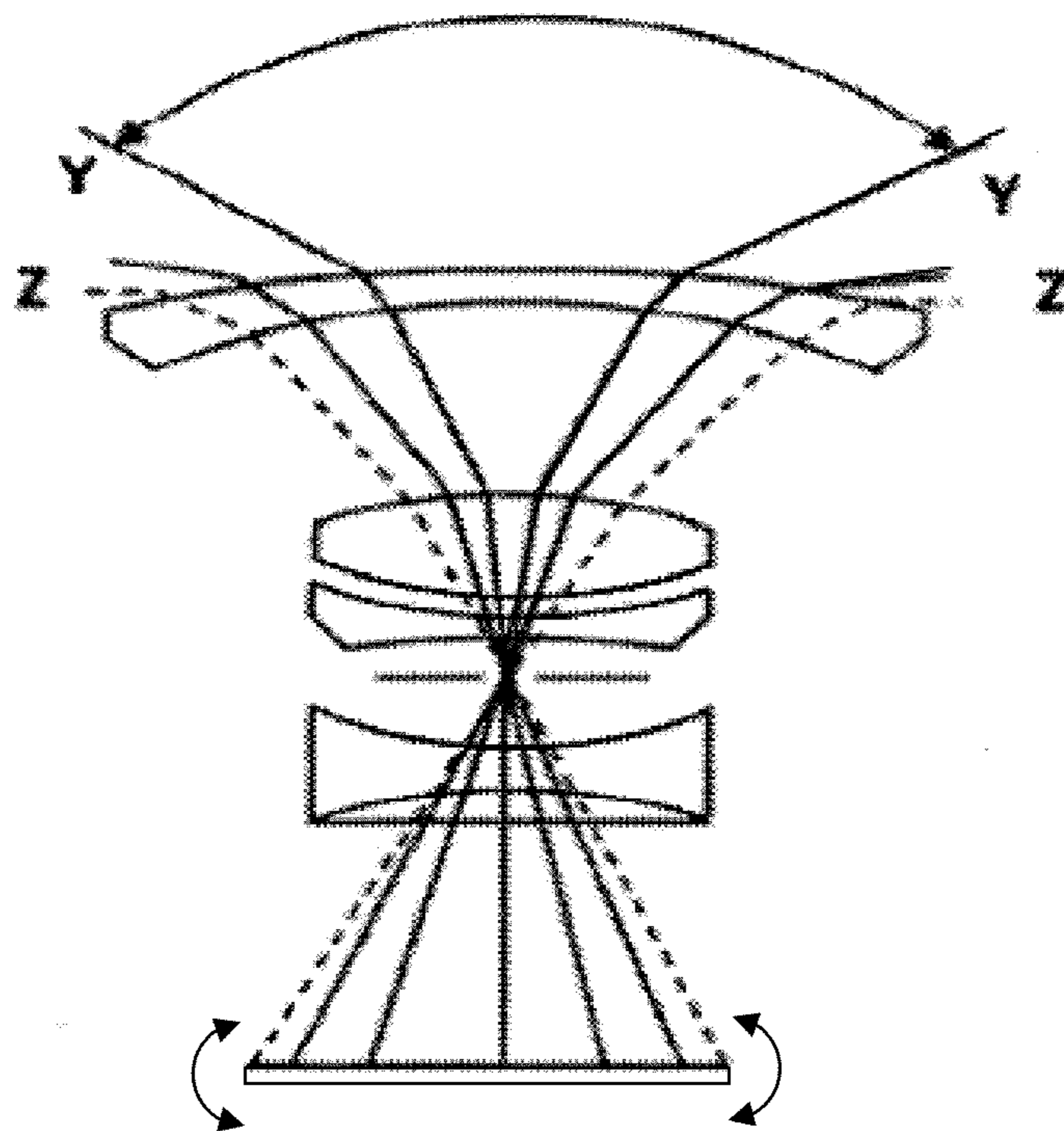


Figure 1 (Prior Art)

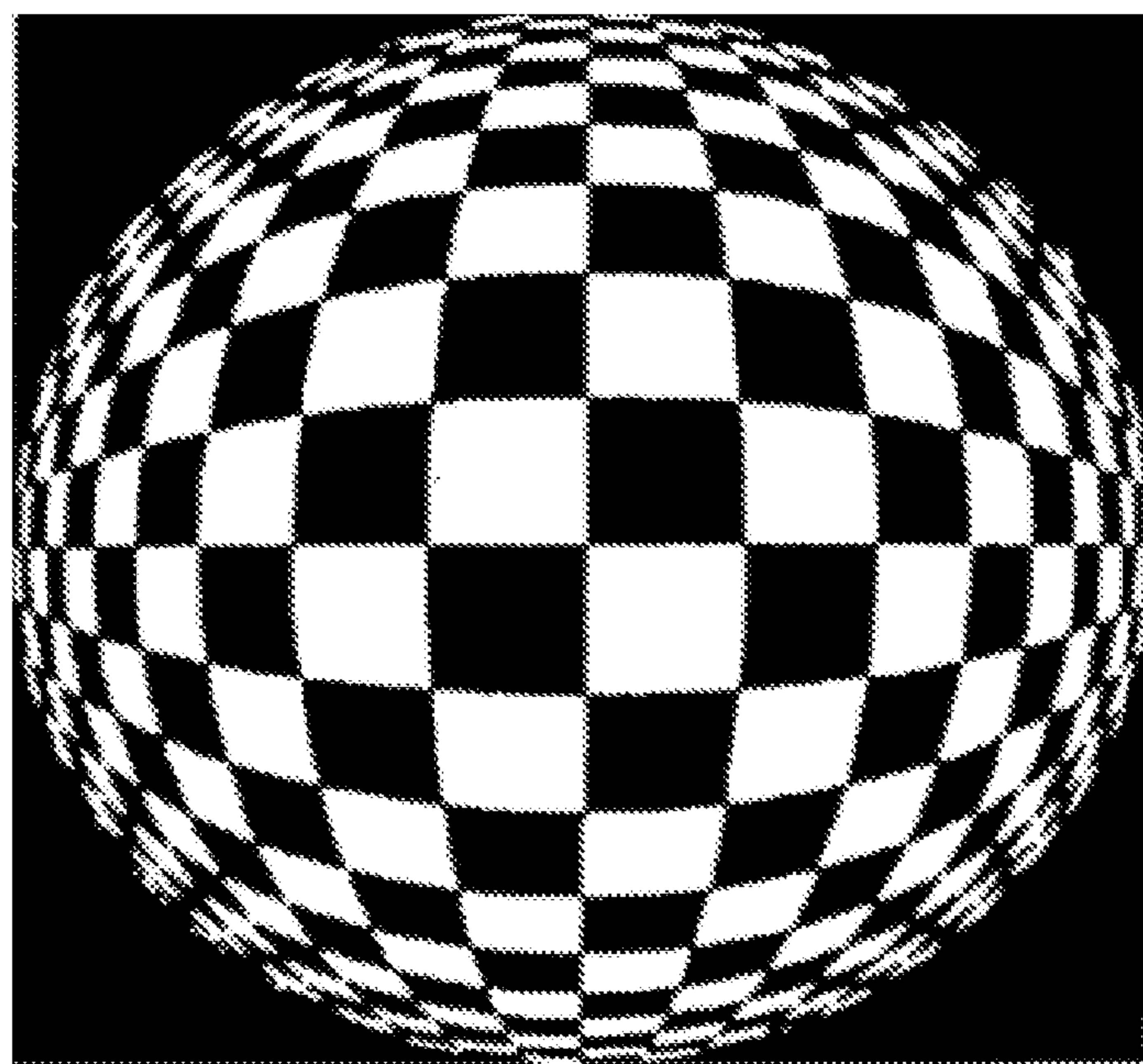
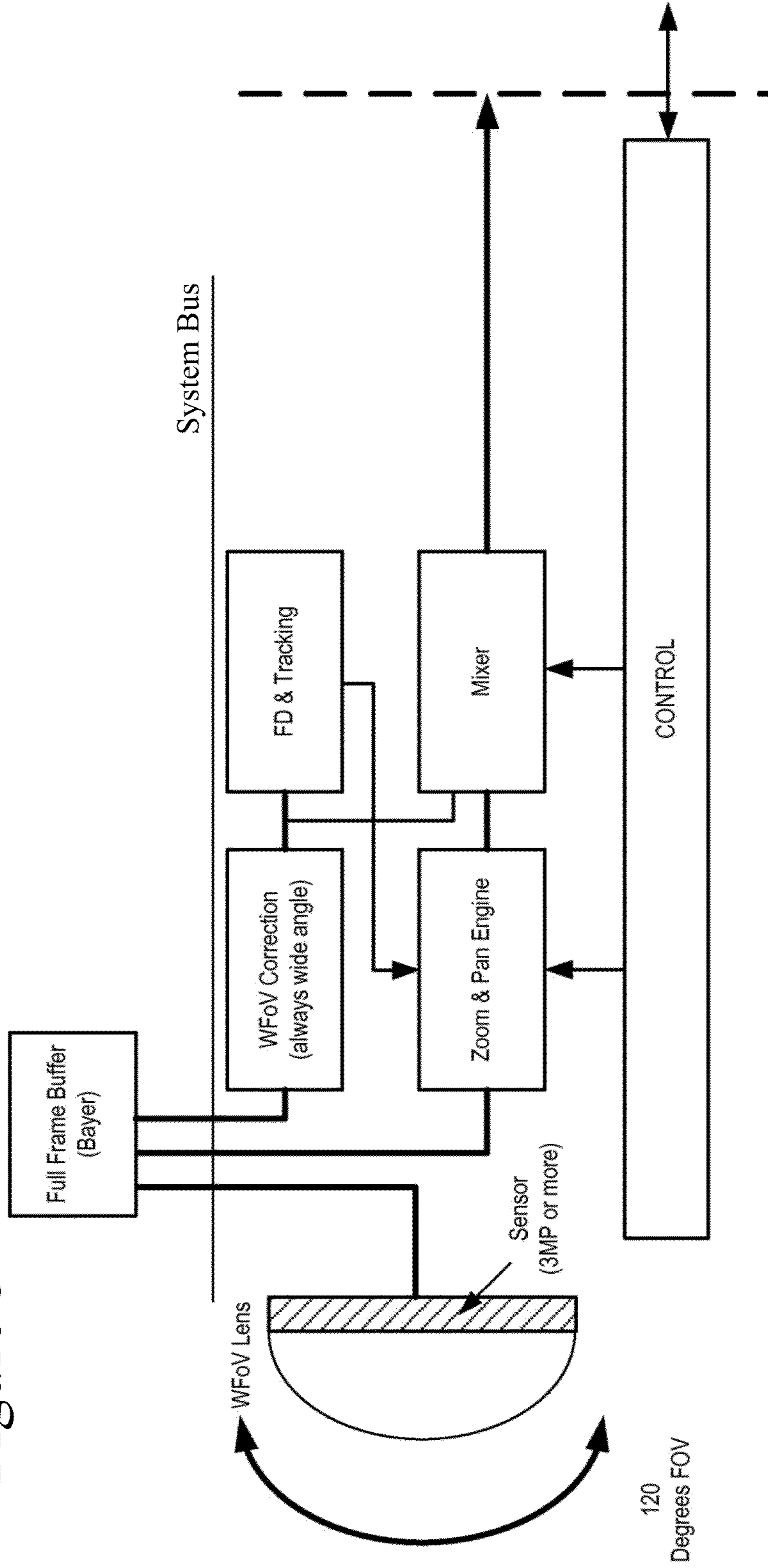


Figure 2 (Prior Art)

Figure 3



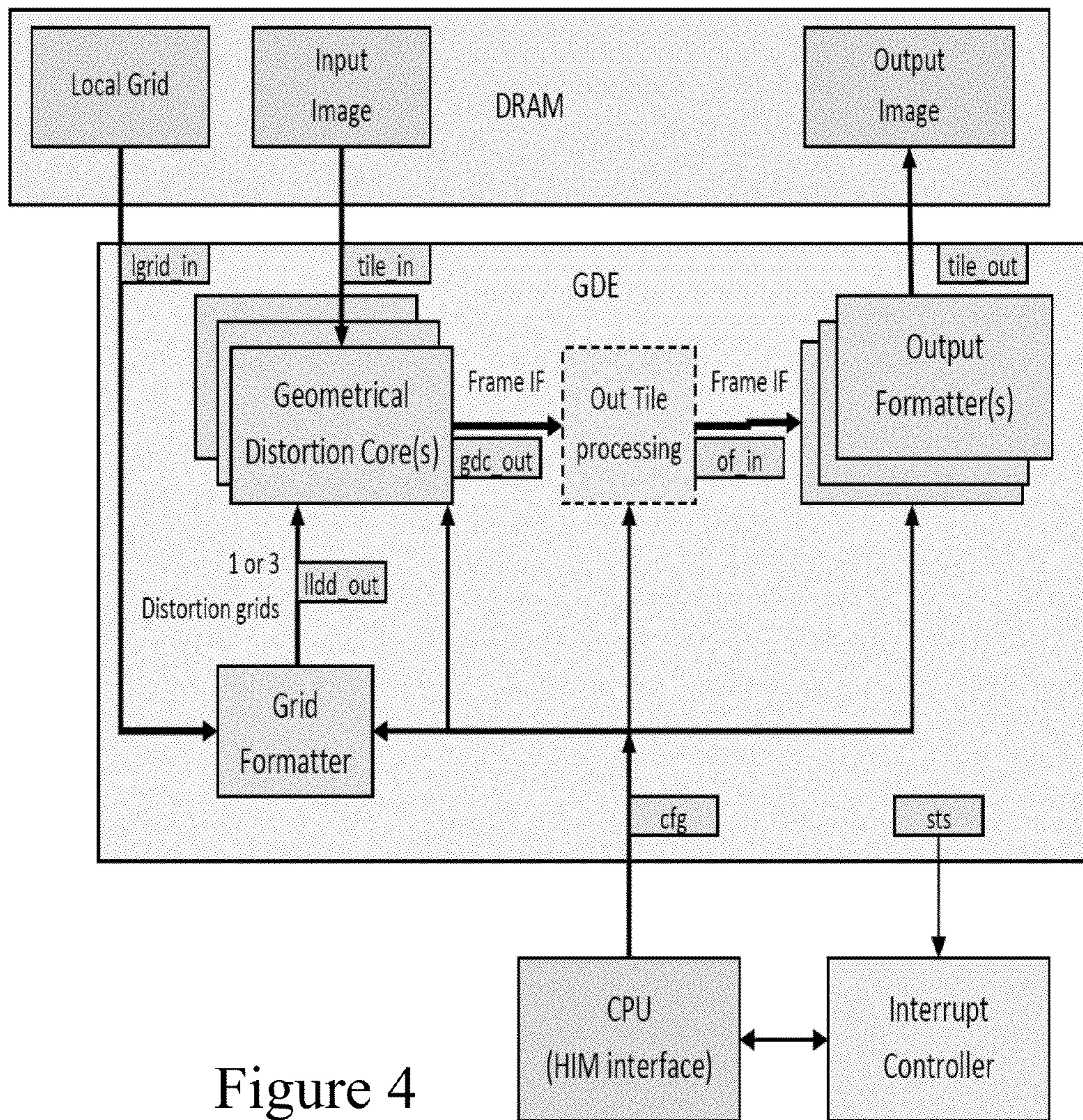


Figure 4

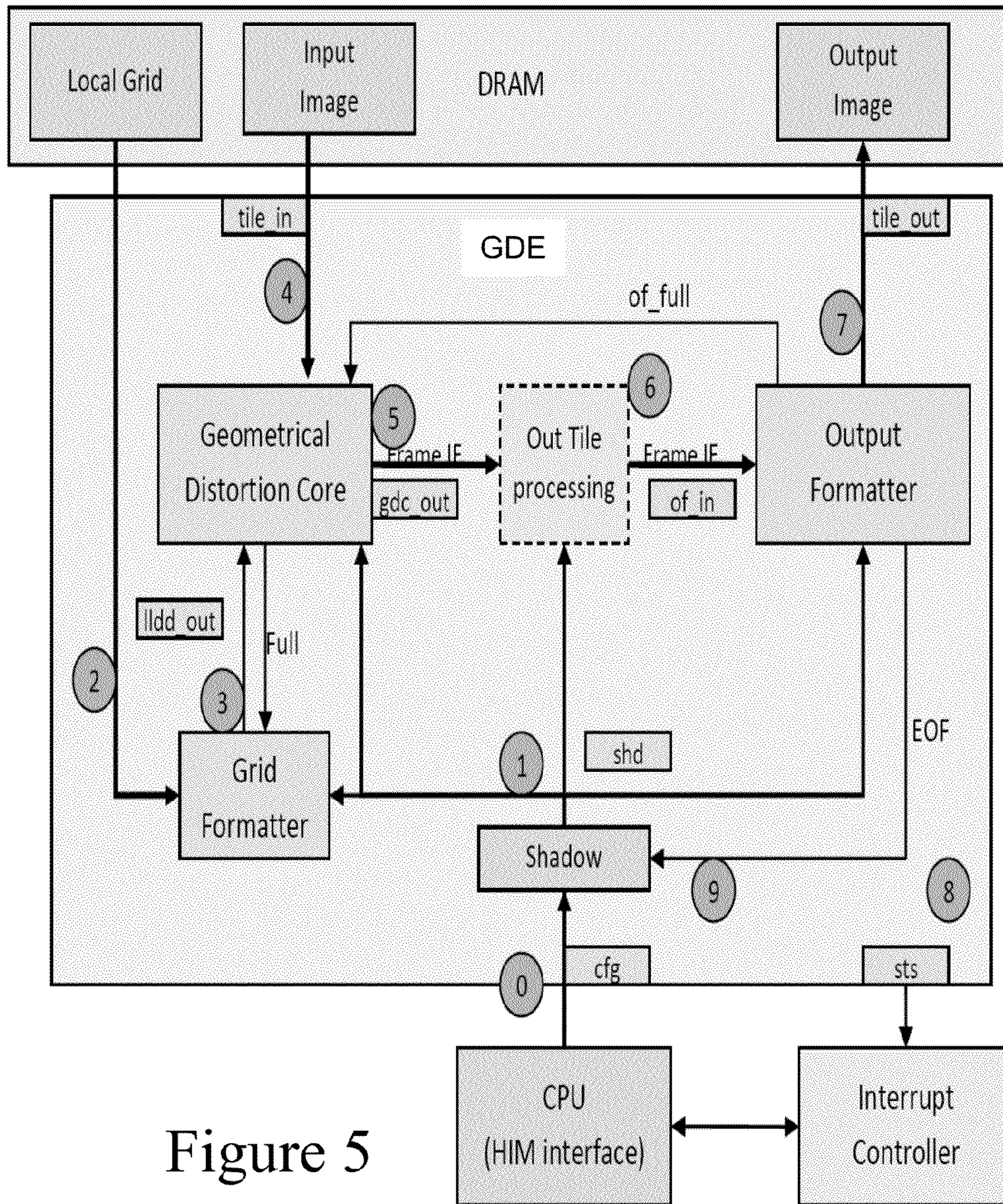


Figure 5

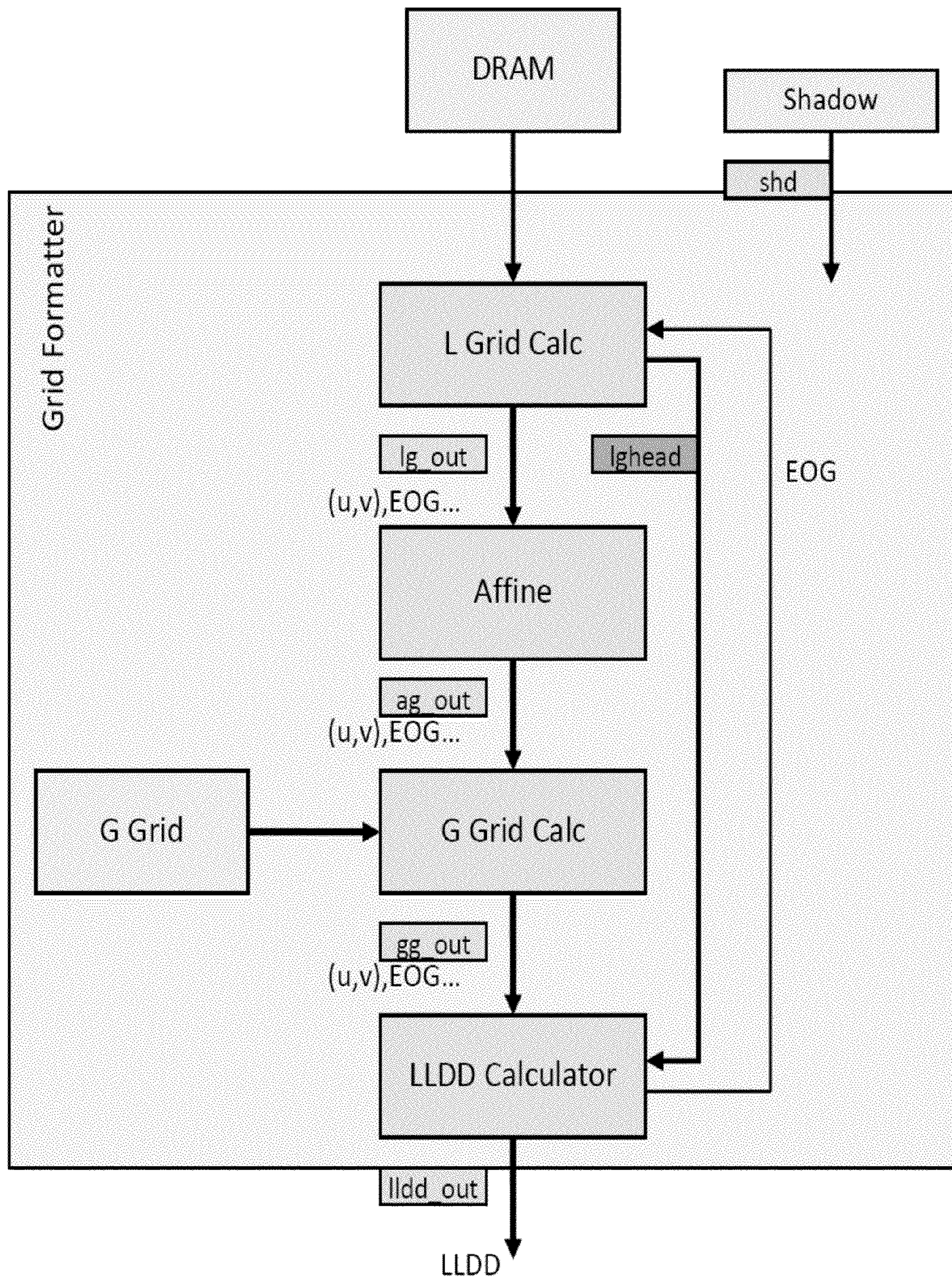


Figure 6

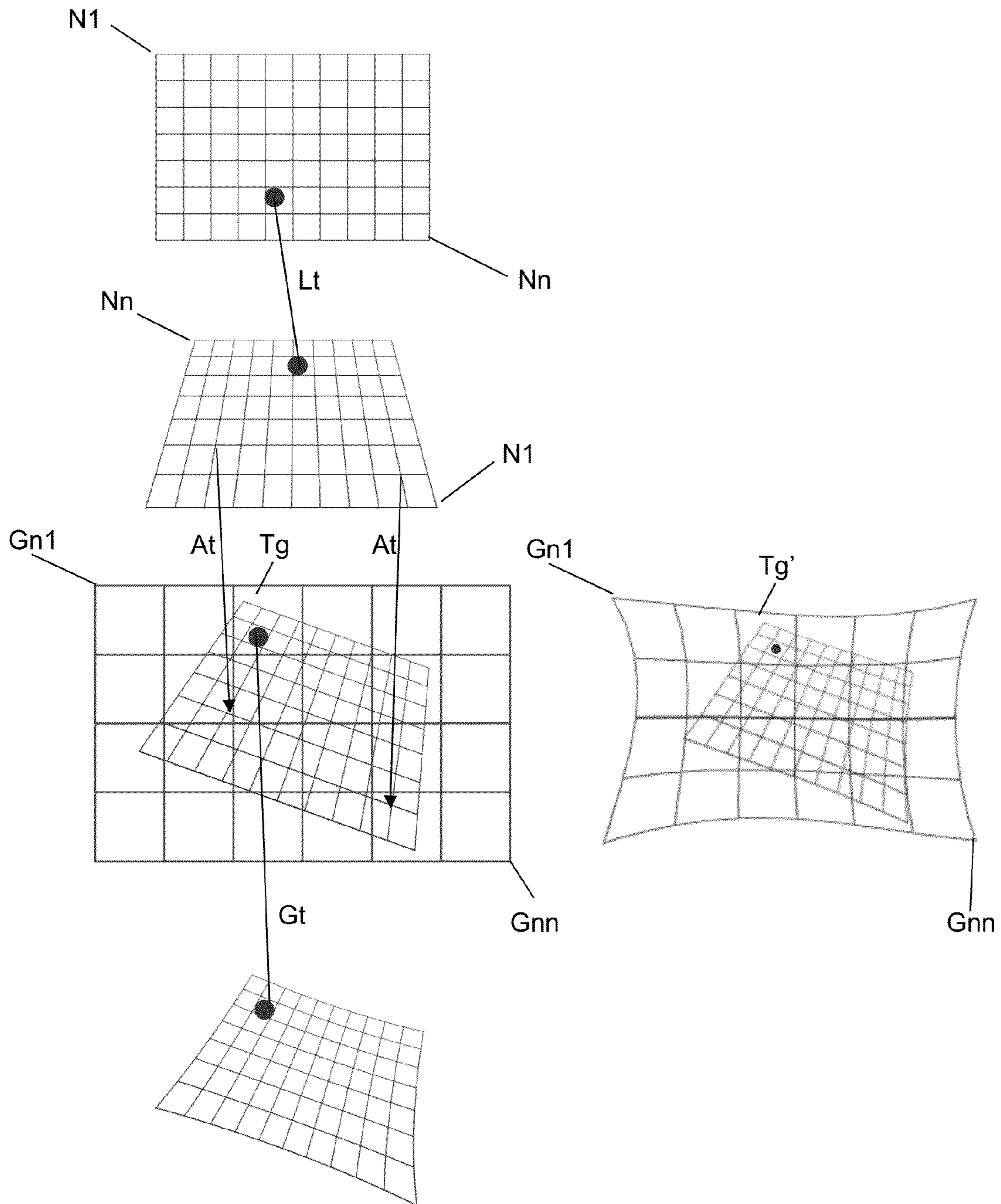


Figure 7(a)

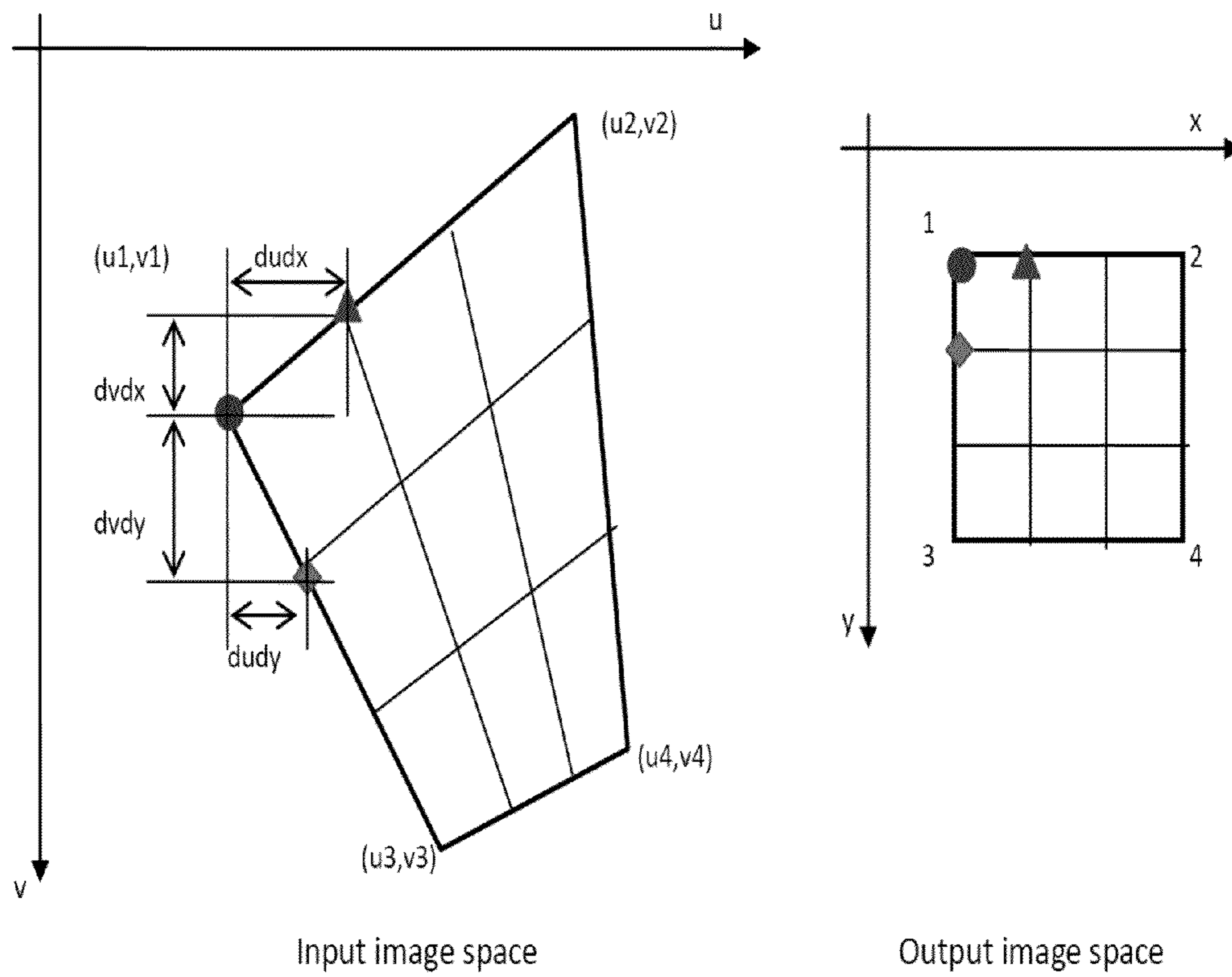


Figure 7(b)

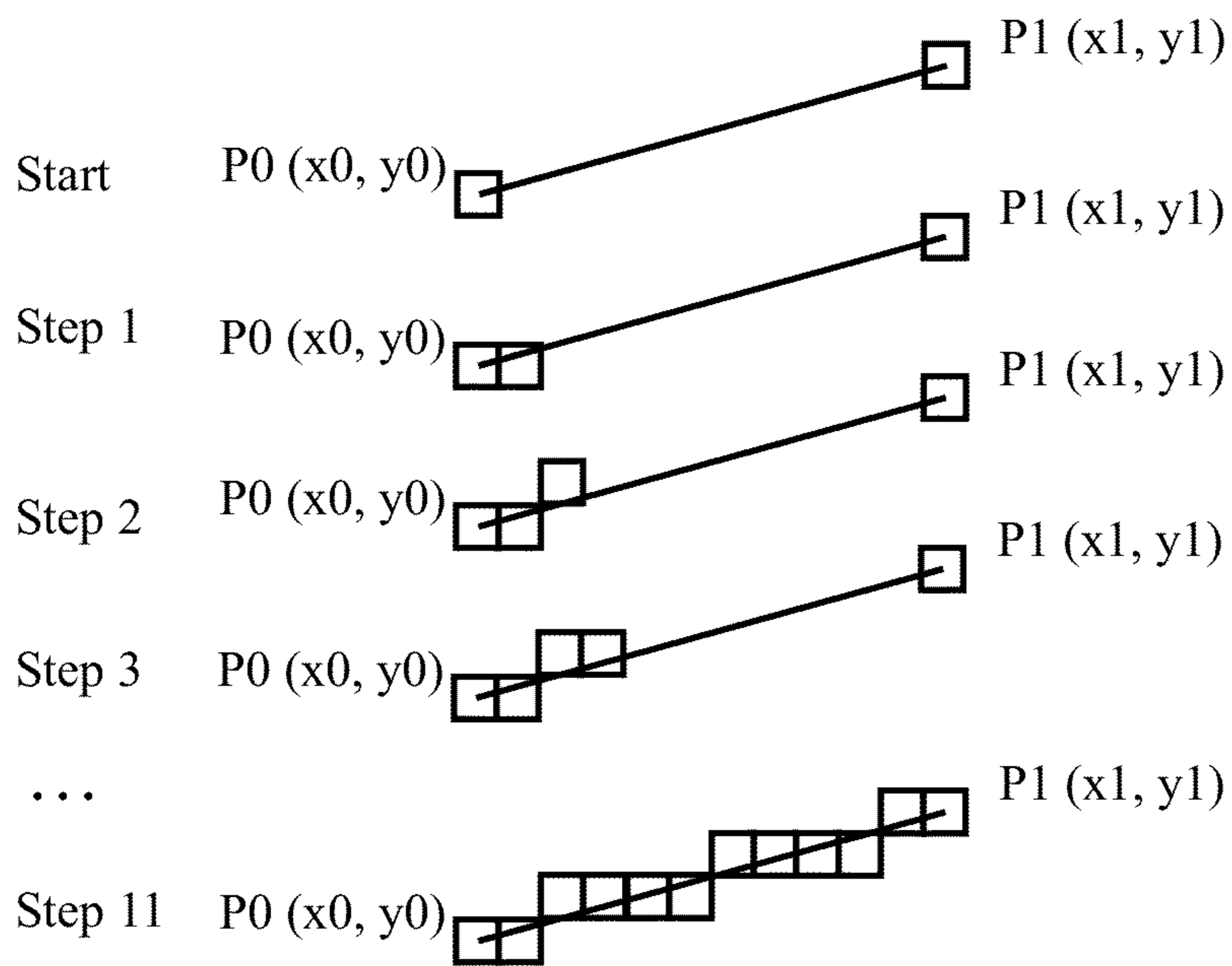


Figure 7(c)

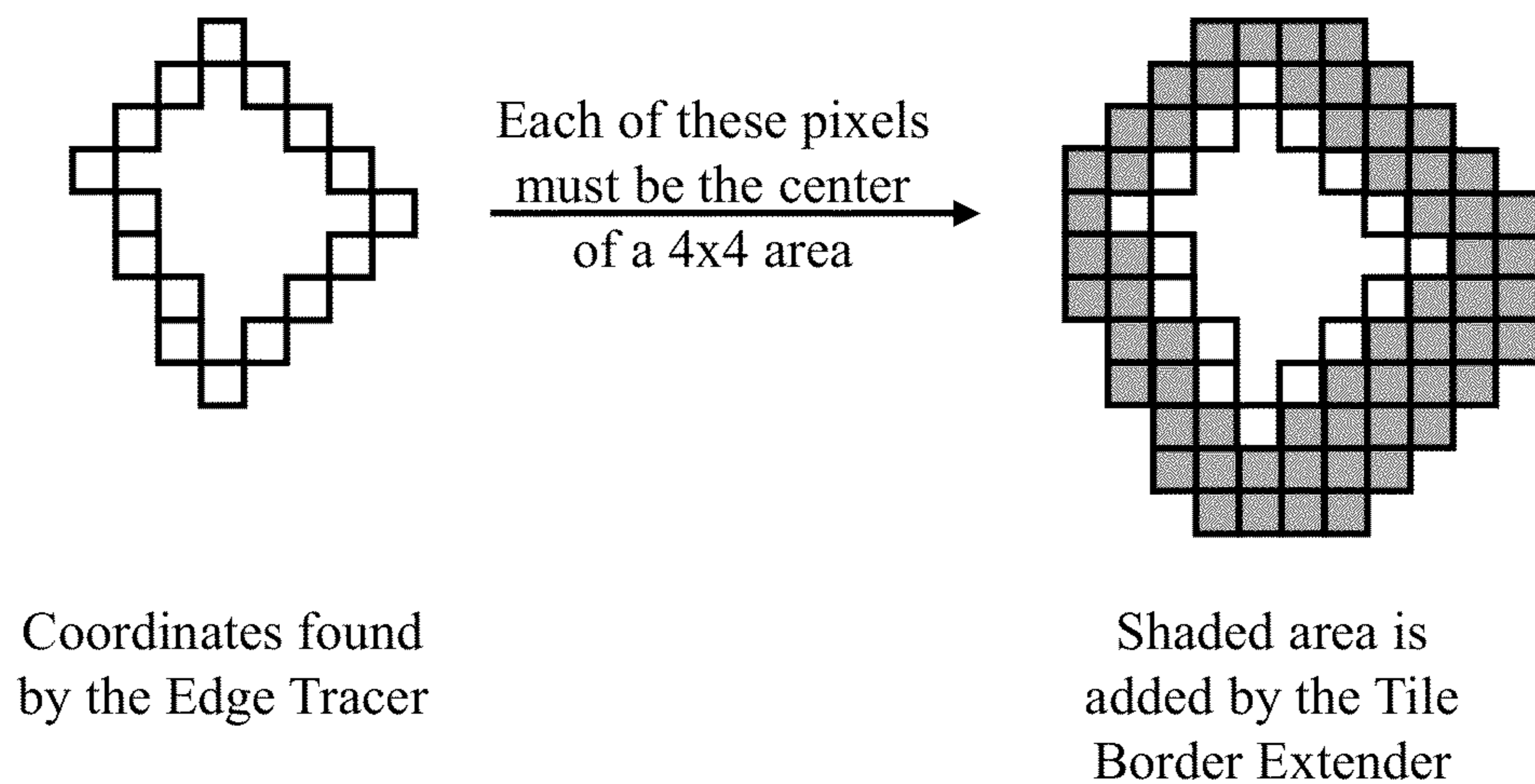


Figure 7(d)

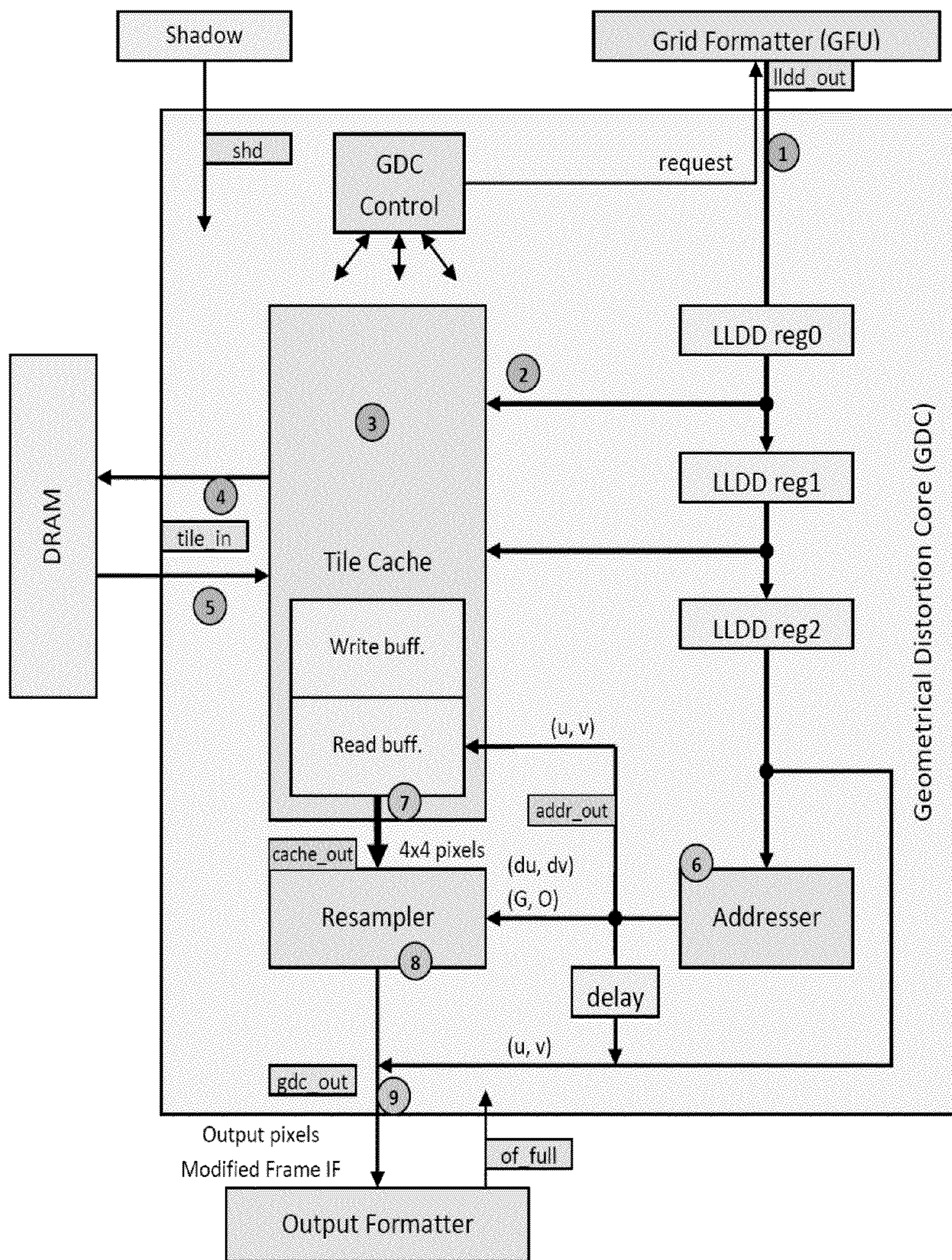


Figure 8

1

**METHOD AND SYSTEM FOR CORRECTING
A DISTORTED INPUT IMAGE**

FIELD OF THE INVENTION

This invention relates to a method and system for correcting a distorted input image.

BACKGROUND

FIG. 1 shows an exemplary Wide Field of View (WFOV) lens system. The lens system shown includes a hemispherical field of view mapped to a planar image sensor for example as shown in FIG. 2.

This mapping can tend to result in variations in acquired image distortion and resolution across the field of view. It is desired to correct for this distortion so that, for example, features such as faces or other objects, especially those located towards the periphery of the field of view, do not appear distorted when displayed.

Separately, WFOV systems can tend to especially introduce heavy and in some cases non-uniform distortion patterns across the field of view so that acquired images (or indeed different colour planes of an acquired image) do not uniformly conform to the ideal mapping shown in FIG. 2. Thus, it is desired to also correct for this type of distortion, as well other types of optical distortion that can tend to occur within digital images.

In a high definition image acquisition device, sometimes enormous amounts of information are received and transmitted across the system bus at high frame acquisition speeds. This places pressure on many processing modules that may be connected to the system bus to ensure their demands on the system bus are within an allocated budget and so do not interfere with other processing. The processing modules themselves may be implemented in certain embodiments with a reduced or minimal hardware footprint so as to minimize device production costs.

U.S. Pat. No. 5,508,734, which is incorporated by reference, discloses a WFOV lens assembly designed to optimize the peripheral regions of the field of view to provide improved resolution matching between the peripheral region relative to a central region, the peripheral region tending to have a lower resolution than the central region.

US 2010/0111440, which is incorporated by reference, by Chai, discloses a distortion correction module which partitions coordinate points in a selected output image into tiles. The output image is an undistorted rendition of a subset of the lens-distorted image. Coordinate points on a border of the tiles in the output image are selected. For each tile, coordinate points in the lens-distorted image corresponding to each selected coordinate point in the output image are calculated. In addition, for each tile, a bounding box on the lens-distorted image is selected. The bounding box includes the calculated coordinates in the lens-distorted image. The bounding boxes are expanded so that they encompass all coordinate points in the lens-distorted image that map to all coordinate points in their respective corresponding tiles. Output pixel values are generated for each tile from pixel values in their corresponding expanded bounding boxes.

BRIEF DESCRIPTION OF THE DRAWINGS

Embodiments are described below, by way of example, with reference to the accompanying drawings, in which:

FIG. 1 illustrates schematically a conventional WFOV lens system;

2

FIG. 2 illustrates a notional distortion pattern introduced by a conventional WFOV lens system;

FIG. 3 illustrates schematically a digital image acquisition device for acquiring and processing a succession of images in accordance with certain embodiments;

FIG. 4 illustrates schematically a geometrical distortion engine (GDE) according to an embodiment;

FIG. 5 illustrates data flow within the GDE of FIG. 4;

FIG. 6 illustrates the structure of the Grid Formatter (GFU) of FIG. 4;

FIG. 7(a) illustrates tile transformation;

FIG. 7(b) illustrates an example of a tile transformed according to an embodiment;

FIG. 7(c) illustrates an example of Bresenham's line algorithm for determining pixels lying along the border of a tile;

FIG. 7(d) illustrates how a tile border produced using the algorithm of FIG. 7(b) is extended; and

FIG. 8 illustrates the structure of a Geometrical Distortion Core (GDC) of FIG. 4.

DETAILED DESCRIPTION OF THE
EMBODIMENTS

Referring to FIG. 3 for illustration of one example embodiment, US published patent application 2011/0216156 and U.S. application Ser. No. 13/077,891 (Ref: FN-369A-US), Ser. Nos. 13/078,970 and 13/084,340, which are assigned to the same assignee and hereby incorporated by reference, disclose digital image acquisition devices including WFOV lens systems. In certain embodiments, distorted WFOV images are read from a sensor via an imaging pipeline which is configured to carry out pre-processing of an image before being read across a system bus into system memory.

Such systems can employ hardware modules or sub-modules also connected directly or indirectly to the system bus for reading successive images stored in system memory from the bus and for processing the image before either returning the processed image to system memory or forwarding the processed image for further processing. The WFOV correction module illustrated by example in FIG. 3 is configured to successively read distorted images or image portions and provide corrected images or image portions to a face detection (FD) and/or tracking module.

In certain embodiments, a system controller controls various hardware modules responsive to, for example, commands received through a control interface from, in certain embodiments, software applications running on the device with which a user may interact. In the example embodiment illustrated in FIG. 3, a zoom and pan module is connected to the controller and this in turn communicates with the WFOV correction module to determine which part of an acquired image will be read from system memory for correction and, for example, displayed on a device viewfinder (not shown) and/or forwarded to the face detection module. In certain embodiments, a mixer module, for example, superimposes boundaries around the faces or other objects which have been detected and/or tracked for display on the device viewfinder.

Part of a correction module footprint in accordance with certain embodiments is cache memory. On the one hand, it may be desirable in certain embodiments to minimize cache size to minimize device cost, yet on the other hand, it may be desirable in certain embodiments to minimize I/O access by hardware modules across the system bus. In certain embodiments, multiple forms of distortion are advantageously corrected, without successively reading from, correcting and writing back to memory an image for each form of distortion to be corrected.

Separately, WFOV lens systems, as well as being incorporated in certain embodiments into hand-held digital image acquisition devices, can be included in other embodiments in devices with various specialist applications, for example, fixed security cameras. In some cases, for example, an over-head camera mounted towards a centre of a ceiling in a room includes in certain embodiments a lens system which is configured to primarily emphasize the circumferential field of view of the room while acquiring relatively little detail in the region immediately below the camera.

When a person walks across such a room, he or she initially moves closer to the camera, but the angle of incidence of his or her face to the camera means the camera view of the face becomes less frontal possibly making it more difficult for the camera to track and/or recognise the person's face. In certain embodiments, in addition to correcting for the distortion introduced by non-linear mapping of the circumferential view of the room onto the planar surface of the acquisition system imaging sensor, advantageous techniques and configurations are provided to adjust either the sensor or the lens angle to improve the view of a target person.

Depending on the nature of the lens assembly, it may be preferable to tilt the lens, rather than the sensor. However, if the lens arrangement includes a large optical assembly, for example, for providing long-range optical quality for security applications, then it could also be desirable to tilt the image sensor assembly, as indicated by the arrows of FIG. 1, to optimize the view of a person's face as they approach the camera. This tilting of the sensor in certain embodiments introduces additional distortion into the image beyond that associated with, for example, a non-linear optical structure of the lens.

As a person approaches a camera, images acquired by the camera of his or her face will tend to become elongated towards the chin and bulbous towards the top of the head. Certain embodiments include advantageous techniques and/or configurations to counter this non-linear distortion of the person's face.

Several different distortions tend to occur in images of a person as he or she walks across a field of view (FOV) towards the lens assembly of a camera, including: (i) a non-linear lens distortion which can be a function of the location within the FOV of the lens; (ii) distortion due to possible relative movement of the lens and sensor surfaces; and (iii) distortion effects in local areas such as faces which vary according to both the vertical and horizontal distance from the camera unit.

Other distortions such as "rolling shutter" distortion can be caused by movement within the field of view while an image is being read from a sensor, such that, without correcting for this distortion, portions of an image can appear wrongly shifted relative to others.

In applications where it may be desirable to flip an acquired image before it is displayed, this can result in distortion and/or be considered as a form of distortion.

In techniques in accordance with certain embodiments, a locally uniform image frame is obtained by dynamically adjusting a mapping between an image sensor surface and rectangular grid regions within a desired view to be presented on a display or otherwise stored for viewing. This mapping can change from frame to frame and indeed within a frame in certain embodiments, and is driven both by the relative position of a moving target relative to the image acquisition device as well as through user interaction with a camera application, for example, determining a size of a region of interest (ROI) within a field of view and/or zooming in on a field of view.

A distortion adjusting engine is provided in certain embodiments which copes with multiple sources of distortion and which can dynamically adjust the overall mapping of pixels from the sensor surface to generate a final rectilinear grid of display pixels on an output display and/or for storing or compressing into a selected video format.

Certain embodiments are particularly useful in and/or configured for security monitoring and/or for monitoring of loved ones, e.g., stay-at-home elderly persons or children.

A basic structure of an engine for handling geometrical distortion within images in a digital image acquisition device according to an embodiment is illustrated by the example provided in FIG. 4. As will be explained in detail below, the Geometrical Distortion Engine (GDE) is capable of effectively removing distortions introduced by, for example, a WFOV lens system. In addition, the GDE illustrated at FIG. 4 may be utilized and/or configured for compensating for distortion caused by camera shake. The GDE of FIG. 4 may also be utilized and/or configured for correcting distortion introduced by a device user through interaction with an application running on or in communication with the acquisition device. Such user-defined distortion may include in certain embodiments an affine transformation, colour transformation and/or image morphing to apply particular effects to the image and indeed sequence of images being acquired by the device.

In certain embodiments, distortion processing on each color plane of an image, for example RGB, YUV or LAB, is performed independently of the others. Accordingly, a GDE for this system may be configured to include a geometrical distortion core (GDC) that is configured to process each color plane separately, so providing greater flexibility at the system level. A single GDC can be configured to process each color plane sequentially, or multiple GDCs (such as the example illustrated at FIG. 4) can process multiple planes of an image separately and/or at the same time.

A grid as used herein may include an array of tiles. Each tile may be defined by its vertices, e.g., the four corners of a rectangular tile, and these may be referred to as nodes. A transformation maps the coordinates of nodes within a grid according to a given distortion to be corrected.

The GDC is configured in certain embodiments to process an input image plane tile by tile under the control of a Grid Formatter Unit (GFU). The GDC fetches input tiles (tile_in) from the DRAM according to the addresses provided by the GFU and processes them, producing the corrected pixels for respective output tiles (gdc_out) in normal raster order.

While information for each distorted tile of the input image may in certain embodiments be read in rectangular blocks from DRAM, each rectangular block bounding a distorted tile, for a heavily distorted input image tile, this can mean reading quite a lot of information from DRAM across the system bus that is not used in mapping the distorted input image tile (tile_in) to the output image tile (gdc_out). In other embodiments, information not used for correcting a given distorted tile is not read from memory into a tile cache (see FIG. 8) of the GDC. In other words, a higher percentage of information that is read from the memory into the tile cache is involved in the correcting of each given distorted tile. In one embodiment, only the information required for correcting a given distorted tile of the input image is read from memory into the tile cache of the GDC. Thus, the nodes defined for each tile to be read from DRAM generally define a polygon which in turn is used to determine the image information read from DRAM for a tile.

In certain embodiments, the distortion function applied by the GDC to each tile may be governed by more than its role in correcting for WFOV lens system distortion. Other distortion

5

effects can be involved such as distortion from camera shake, user defined distortion and/or lens-sensor misalignment (sensor tilt).

As will be described in more detail in relation to FIG. 6, the GFU combines in certain embodiments local grid information taken from DRAM, an affine transformation matrix and global grid information, and may also produce Low Level Distortion Descriptors (LLDD) for each tile of the grid from within a sensed image which is to be processed by the GDE or each GDC. These descriptors may be employed by the GDE or each GDC to read correct image tile information from memory and to correct the image tile.

In the present description, a Local Grid may relate to an area of interest within a field of view where the image is to be corrected, for example, for subsequent display. So for example, if in an image stream, a face detector (FD) such as that illustrated in FIG. 3, is configured to detect a face region within a field of view, an application fed from the FD may be configured in certain embodiments to decide that a region, e.g., having polygonal shape such as that of a rectangle, bounding this face represents a region of interest. The coordinates defining this region are then written to a "Local Grid" region of DRAM and this region will be processed by the GDE. Thus, in this embodiment, for a given frame, at least one Local Grid is stored in memory defining a region of the field of view which is to be corrected. As a face moves across the field of view of the camera, the Local Grid can be shifted from frame to frame. Equally, if more than one face is detected, more than one Local Grid may be stored in memory and each Local Grid may be processed by the GDE in turn. As such, the description of the processing of each Local Grid can be treated independently.

The corrected grids of the complete image could for example be displayed superimposed over the remaining portions of the image, so that for example faces which are detected at the extreme edges of the field of view of a WFOV lens system can be displayed undistorted.

The Affine Transformation enables the GDC to correct either, for example, for movement from frame to frame or indeed to compensate for changes or misalignment between lens and image sensor (Global Affine); or for example, distortion caused by rolling shutter (Local Affine). Thus, in the case of local affine transformation, the mapping of node locations from one portion of the Local Grid of an input image to the output image could be different from the mapping from another portion of the Local Grid. This may be implemented by specifying sequences of nodes for which given transformations apply.

The Global Transformation is in general fixed for a given lens. For a typical WFOV lens, the transformation takes into account the deviation caused by a given lens away from a nominal mapping of field of view to an image sensor such as that illustrated in FIG. 2. For a zoom lens, the Global Transformation may be fixed for a given focal length. For a lens such as may be configured for use within a security camera, where the image sensor can rotate relative to the lens system, the Global Transformation may be fixed for a given angle of the image sensor to the lens system. This mapping is certain embodiments stored within the GFU, where it may be only rarely updated or, except in the case of a zoom lens, it may be at least unlikely to be updated on a real-time frame-by-frame basis as an image stream is captured.

Referring back to FIG. 4, an output formatter is configured in certain embodiments to take corrected image tiles (of_in) and write these into Output Images in bursts back to the DRAM memory.

6

Extra "Out Tile" processing blocks can be inserted in certain embodiments between the GDC and the output formatter. In certain embodiments, the output format of each GDC is in a standard frame format, e.g., so that each tile output by the GDC can be treated as a separate image, meaning that a "Out Tile" processing block that has a frame interface input/output can be inserted between the GDC and output formatter. The extra processing blocks can be blocks that process a pixel deep image stream, for example, gamma correction, colour enhancement and/or high dynamic range processing. They can also be blocks where a second image source is used, for example, for alpha blending.

Referring now to FIG. 5, the operation of a GDE in accordance with certain embodiments is described in an example process for a given image plane:

0 The CPU programs the GFU and the other blocks.

1 When the GDE block is enabled, the configuration from the CPU is copied into internal shadow registers via the cfg interface. The main purpose of the shadow registers bank is to provide constant configuration inputs to the internal GDE blocks during processing of a given image frame while allowing the CPU to prepare the configuration for the next image frame. As such the contents of the shadow registers are in general stable for the whole processing time of a frame.

2 Referring to FIG. 6, the GFU fetches a local grid header from the DRAM. The header contains information about local grid location in the output image space, grid dimension expressed in number of nodes, and the single cell (tile) size. The output image grid is rectangular in this example and so this information is enough to describe the whole local grid in the output image space. The header is followed by the nodes containing coordinates in the input image space and variations in those coordinates describe the distortion associated with the local grid. So referring to the example of FIG. 7(a), the local grid transformation (Lt) defines for each node of the Local Grid, the change in node coordinate information. In the example of FIG. 7(a), the local grid comprising nodes N1 . . . Nn undergoes a flip and perspective transformation and so could compensate for varying rotation of the image sensor relative to the lens assembly such as illustrated in FIG. 1 as well as for simulating or compensating for a mirror view. So we can see node N1 moves from the top left to the bottom right and vice versa with node Nn, and that the transformed top of the local grid becomes more compressed than the bottom. The effect of the local grid transformation on the geometry and location of a specific tile is also illustrated.

Also, the local grid transformation can compensate for varying distortion caused by changes in perspective for different local regions within an input image—particularly in the case of WFOV systems. Thus, the local grid can help to compensate for the greater degree of distortion found in faces at the edge of a wide field of view vis-à-vis those located (or detected) at the centre of the field of view.

Values from the local grid header are used by L Grid Calc to setup registers responsible for accessing the local grid information from DRAM. After this, the GFU starts to read local grid node coefficients from DRAM one by one. The transformed coordinates for the grid nodes are then passed to an Affine block (if enabled). In the embodiment, the Affine block multiplies input node coordinates u, v by a 2×3 matrix comprising coefficients $a1 \dots a6$ of the Affine Transformation (At) in order to produce output coordinate values u', v' :

$$\begin{bmatrix} u' \\ v' \end{bmatrix} = \begin{bmatrix} a1 & a2 \\ a3 & a4 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} + \begin{bmatrix} a5 \\ a6 \end{bmatrix}$$

The values of those matrix coefficients $a1 \dots a6$ are stored in the registers internal to the GFU. These internal GFU registers holding coefficient values can be programmed two-fold: in a first mode, Global Affine Transform mentioned above, they can be programmed by the CPU before the start of the frame processing and their values are kept constant for all local grids of a whole frame; and in the second mode, Local Affine Transform, values of the shadow registers are read from DRAM together with a node index that indicates when a new set of affine transformation coefficients must be loaded. For example, if a first set of node coefficients is loaded together with an index 100, this transform is applied to the nodes 0 to 99 and before node 100 is processed a new set of transformation coefficients is loaded from DRAM and applied to the subsequent nodes until the next change is indicated. As mentioned above, the second mode allows for dynamic transformation updates and correction for example, of rolling shutter distortions together with camera shake compensation. Thus, it will be seen that in this example, the Affine Transformation comprises a formulaic transformation of node coordinate locations from the local transformation (Lt). In the present example show in FIG. 7(a), the affine transformation (At) comprises a global affine transformation rotating the entire grid and so could compensate for a rotational misalignment in a plane parallel to the plane of the image sensor between lens and image sensor.

The coordinates that are produced by the Affine block of FIG. 6 are then passed to the global correction block G Grid Calc. This block applies warping distortion to the input local and/or affine transformed node coordinates with the distortion defined by means of a regular grid of nodes (G in FIG. 7(a)) with nodes distributed regularly in the input coordinates space and the values stored in the nodes point to locations on the sensor image space. This provides mapping from the regular to distorted coordinate system with intermediate coordinate values (not belonging to the grid nodes) obtained using bi-cubic interpolation. The values of the coordinates of the global grid nodes are stored in the internal registers and can be updated after the end of each frame to allow for dynamic changes to the correction for lens distortion involved in an embodiment including a zoom lens where distortion changes shape with change of the focal length. The final node coordinate values from the global transformation are passed to the LLDD Calculator input queue.

Again referring to the example of FIG. 7(a), the global transformation (Gt) comprises a mapping of node coordinates $Gn1$ to Gnn of a Global Grid for example to take into account lens distortion. For a given node coordinate after Affine Transformation (At) and or Local Grid transformation (Lt), G Grid Calc finds the nodes of the Global Grid surrounding that location and interpolates the mapping of those nodes of the Global Grid tile to the node of the local and/or affine transformed node of the local grid to determine the Global Transform (Gt) of that node location.

In the example illustrated at FIG. 7(a), for nodes within the tile Tg of the global grid, the mapping of the coordinates of the tile Tg to the transformed file Tg' is interpolated and applied in certain embodiments to local and/or affine transformed node coefficients to finalise the transformation of the original node coordinates. Thus for a node N , the complete transformation becomes $Gt(At(Lt(N)))$.

When an LLDD Calculator in accordance with certain embodiments input queue contains a number of nodes equal to grid width+2 (a full tile), it uses them to prepare an LLDD descriptor that contains a full definition of the input tile. The definition contains location of the tile on the sensor image and partial differences that will be used by an address calculator (see, e.g., FIG. 8) to calculate locations of pixels belonging to this particular tile. The complete LLDD descriptors are loaded to the LLDD FIFO (lidd_out).

3 Referring back to FIG. 5, the GFU fills the LLDD FIFO of the GDC with descriptor data for each tile to be processed.

4 The GDC fetches the input image tile by tile, with a new tile for every LLDD FIFO entry.

5 The GDC processes each tile and outputs the corrected tiles in frame interface format. A backpressure signal path from the output formatter to the GFU enables the GFU to stall the GDC if the output formatter is full.

6 Optional processing algorithms can be applied on the GDC corrected tiles.

7 The output formatter writes the corrected tiles (tile_out) of the output image into the memory.

8 When processing of a given Local Grid tile is completed and when the frame is completed, the output formatter signals this using an interrupt signal provided through a status interface (sts) to an interrupt controller.

9 If the GDE is still enabled when the frame is completed (EOF), the shadow register values are updated for the next frame.

Referring to FIG. 6, as indicated above, the GFU in this example combines basic local grid information obtained by the L Grid Calc block from DRAM, affine transformation matrix and the global distortion grid information stored within the block G Grid and obtained from the Shadow Registers and generates the LLDD descriptor information for each tile.

When the GFU is enabled, an L Grid Calc block in accordance with certain embodiments starts reading the local distortion grid (defining Lt in FIG. 7(a)) from the Local Grid information in DRAM. A frame includes at least one local grid, unless whatever application is running in the acquisition device has determined that there is no particular region of interest where distortion correction is to be applied, such as for example, when no faces or other objects of interest have been detected or are being tracked. Multiple local grids may be stored one after another in the memory. A local grid may include a header followed by the nodes of the grid that contain point coordinates in the distorted image space.

An Affine transformation block in accordance with certain embodiments applies a user defined affine transformation (see, e.g., At in FIG. 7(a)) to the (u, v) coordinates produced by the L Grid Calc block. The Affine block performs an affine transformation on the (u,v) coordinates from the L Grid Calc block. As described at step 2 above, the Affine block in this illustrative, non-limiting example has two operating modes where a) the affine coefficients are taken from the internal GFU registers corresponding to the shadow registers meaning that they are constant through the frame; and b) the coefficients are taken from the internal GFU registers and can change during the frame.

A G Grid calculation block according to certain embodiments calculates the final distorted grid including performing spline interpolation based on the global grid points (see, e.g., Gn in FIG. 7(a)) obtained from G Grid.

When reading the last node of the last tile of the current local grid, L Grid Calc asserts an End of Grid (EOG) flag. The grid coordinates in input space (u,v) and output space (x,y) together with the EOG flag are sent to the next block in the

pipe—in this example Affine. The next blocks in the pipe (Affine, Global Grid calculation) may use the same interface, meaning that the Affine or the Global Grid Calculator blocks can be swapped or removed from the pipeline. The (u,v) coordinate may be processed by the Affine and Global Grid calculator. Other image fields in the header are passed down the pipeline unchanged.

The final distortion descriptors for each tile of the grid are calculated by an LLDD Calculator. The LLDD Calculator block combines the header information provided on an lghead interface in certain embodiments with the descriptor fields, and sends them on a lldd_out interface. The L Grid Calc block does not start processing a new grid in this example until the LLDD Calculator block signals with an EOG signal that the last tile of the current grid is processed. This ensures that the signals on the lghead interface are constant for all tiles of a local grid.

FIG. 7(b) shows a tile in the output (right) and the input (left) image space. For exemplary purposes, the tile contains 4×4 pixels. The LLDD Calculator gets the coordinates of the four corners (u1,v1) to (u4,v4) and calculates the partial differences (dudx, dvdx, etc) needed by an addresser within the GDC for the linear interpolation of each of one or more sets of pixel (u,v) coordinates. As indicated above, knowing the various transformations that may be used to compensate for camera movement and user determined distortion, the LLDD calculator in certain embodiments can determine an area of input image space defined by (u1,v1) . . . (u4,v4) to bounded output image space defined by nodes 1, 2, 3, 4.

When interpolating input image data to calculate output image values, data may be used for points outside the boundary defined by the vertices (u1,v1) . . . (u4,v4).

The LLDD Calculator could therefore be used to determine the memory addresses corresponding to the tile border and to extend the memory addresses around this border for each tile using, for example, a variation of Bresenham's line algorithm.

FIG. 7(c) shows an example of the steps performed by an LLDD Calculator module in accordance with certain embodiments. Here, the module takes the first and last point of each edge (u1,v1 and u2,v2; u2,v2 and u4,v4 etc) and computes (one by one) the coordinates of the pixels located on the line that is described by those 2 points.

Each of one or more sets of edge (x,y) coordinates is analyzed and the minimum and maximum x coordinates of each line in DRAM from which tile information is to be read by the GDC are stored in respective memories Max and Min. The y coordinate represents the memory address. After an edge tracer within LLDD Calculator finishes all 4 edges of a tile, it sends a ready indication to a tile border extender module within LLDD Calculator. This extender module extends the start/end coordinates produced by the edge tracer. The extension is used in this embodiment, wherein a 4 pixel×4 pixel area around each pixel and the coordinates computed by the edge tracer are changed to include all the pixels to be used. The extender module reads the 2 memories Max and Min and determines the final start/end coordinates of the pixels of each line of the tile as illustrated in the example shown in FIG. 7(d).

Thus, the above LLDD Calculator takes transformed node coordinates for a tile provided by G Grid Calc (or indeed any of one or more of the previous transformations) and provides non-rectangular strips of memory addresses running from Min to Max for each line of the input image for a tile to be read from memory by the GDC when correcting the input image.

In an alternative implementation, rather than providing the actual memory addresses to be read by the GDC, an LLDD

Calculator in certain embodiments provides the tile descriptor information illustrated in FIG. 7(b). The tracing/extending functionality described above for converting this descriptor information to memory addresses may be implemented within the GDC, e.g., as set forth below.

FIG. 8 illustrates a block diagram of a Geometrical Distortion Core (GDC) in accordance with certain embodiments. Sub-blocks of the GDC may include:

Geometric Distortion Core (GDC) Control—the main control sub-block

LLDD Registers—Low Level Distortion Description Registers. Each time the LLDD for a new tile is requested from the GFU, these registers are shifted. There are two such registers as there can be data for up to three tiles in the pipe at one time.

Tile Cache—a double buffered cache which contains a Burst Calculation module (not shown) which calculates burst accesses to fill the tile cache and load the cache with data from the DRAM.

Addresser—for each pixel in the output tile (in raster order), the Addresser calculates:

coordinates of the 4 pixel×4 pixel window from the Tile cache

sub-pixels (dx,dy) for the Resampler

color offset and gain for the resampler output

Resampler—a bicubic resampler which produces an interpolated pixel value from a 4×4 pixel input.

Referring to the steps indicated in the illustrative example of FIG. 8, the GDC may operate in accordance with the following operations to prepare input tile and load to Tile Cache:

1 The GDC control block requests a new distortion descriptor LLDD for a tile. In this example, it is assumed that LLDD Calculator provides descriptor information as shown in FIG. 7(b) from the GFU and memory addresses are calculated by the Tile Cache;

2 Once the pipeline allows a new tile to be prepared, the Burst Calculation module within the tile cache starts working on the LLDD descriptor data from the GFU;

3 The Burst Calculation module calculates one by one the burst requests for the tile;

4 The Burst Calculation module requests the burst data from the DRAM based on LLDD information;

5 The Burst data is received from the DRAM and written to the Tile cache.

Process Tile:

6 For each output tile pixel, the addresser calculates the address of each 4 pixel×4 pixel block and the parameters for the Resampler

7 The 4×4 pixels window is fetched from the Tile Cache

8 The Resampler calculates the resampled output pixel

9. The signals for the gdc_out interface are assembled together in this example. It contains:

pixel data from the Resampler

Frame control signals from the addresser

Tile control signals from the LLDD FIFO

Output local grid information from the lghead register

An efficient mechanism is provided in certain embodiments for performing complex distortion compensation on an input image in a processor and memory in an efficient manner with relatively low or even minimal demands on the system bus.

Advantageous correction modules are provided herein to address distortion problems in images captured by various types of digital image acquisition devices including digital still cameras, video cameras, camera-phones, camera modules, web cameras, and other camera-enabled devices. All

references cited herein are incorporated by reference, including the following as describing camera devices and features in accordance with alternative embodiments:

U.S. Pat. Nos. 7,224,056, 7,683,468, 7,936,062, 7,935, 568, 7,927,070, 7,858,445, 7,807,508, 7,569,424, 7,449,779, 7,443,597, 7,768,574, 7,593,636, 7,566,853, 8,005,268, 8,014,662, 8,090,252, 8,004,780, 8,119,516, 7,920,163, 7,747,155, 7,368,695, 7,095,054, 6,888,168, 6,583,444, and 5,882,221, and

US published patent applications nos. 2012/0063761, 2011/0317013, 2011/0255182, 2011/0274423, 2010/0053407, 2009/0212381, 2009/0023249, 2008/0296717, 2008/0099907, 2008/0099900, 2008/0029879, 2007/0190747, 2007/0190691, 2007/0145564, 2007/0138644, 2007/0096312, 2007/0096311, 2007/0096295, 2005/0095835, 2005/0087861, 2005/0085016, 2005/0082654, 2005/0082653, 2005/0067688, and

U.S. patent application No. 61/609,293, and

PCT applications nos. PCT/US12/24018 and PCT/US12/25758.

Components of MEMS actuators in accordance with alternative embodiments are described at U.S. Pat. Nos. 7,972,070, 8,014,662, 8,090,252, 8,004,780, 7,747,155, 7,990,628, 7,660,056, 7,869,701, 7,844,172, 7,832,948, 7,729,601, 7,787,198, 7,515,362, 7,697,831, 7,663,817, 7,769,284, 7,545,591, 7,792,421, 7,693,408, 7,697,834, 7,359,131, 7,785,023, 7,702,226, 7,769,281, 7,697,829, 7,560,679, 7,565,070, 7,570,882, 7,838,322, 7,359,130, 7,345,827, 7,813,634, 7,555,210, 7,646,969, 7,403,344, 7,495,852, 7,729,603, 7,477,400, 7,583,006, 7,477,842, 7,663,289, 7,266,272, 7,113,688, 7,640,803, 6,934,087, 6,850,675, 6,661,962, 6,738,177 and 6,516,109; and at

US published patent applications nos. 2010/030843, 2007/0052132, 2011/0317013, 2011/0255182, 2011/0274423, and

U.S. unpublished patent application Ser. Nos. 13/302,310, 13/247,938, 13/247,925, 13/247,919, 13/247,906, 13/247,902, 13/247,898, 13/247,895, 13/247,888, 13/247,869, 13/247,847, 13/079,681, 13/008,254, 12/946,680, 12/946,670, 12/946,657, 12/946,646, 12/946,624, 12/946,614, 12/946,557, 12/946,543, 12/946,526, 12/946,515, 12/946,495, 12/946,466, 12/946,430, 12/946,396, 12/873,962, 12/848,804, 12/646,722, 12/273,851, 12/273,785, 11/735,803, 11/734,700, 11/848,996, 11/491,742, and at

USPTO-Patent Cooperation Treaty applications (PCTS) nos. PCT/US12/24018, PCT/US11/59446, PCT/US11/59437, PCT/US11/59435, PCT/US11/59427, PCT/US11/59420, PCT/US11/59415, PCT/US11/59414, PCT/US11/59403, PCT/US11/59387, PCT/US11/59385, PCT/US10/36749, PCT/US07/84343, and PCT/US07/84301, which are all incorporated by reference.

All references cited above and below herein are incorporated by reference, as well as the background, abstract and brief description of the drawings, and U.S. patent application Ser. Nos. 12/213,472, 12/225,591, 12/289,339, 12/774,486, 13/026,936, 13/026,937, 13/036,938, 13/027,175, 13/027,203, 13/027,219, 13/051,233, 13/163,648, 13/264,251, and PCT published application WO2007/110097, and U.S. Pat. Nos. 6,873,358, and RE42,898.

The following are also incorporated by reference as disclosing alternative embodiments:

U.S. Pat. Nos. 8,055,029, 7,855,737, 7,995,804, 7,970,182, 7,916,897, 8,081,254, 7,620,218, 7,995,855, 7,551,800, 7,515,740, 7,460,695, 7,965,875, 7,403,643, 7,916,971, 7,773,118, 8,055,067, 7,844,076, 7,315,631, 7,792,335, 7,680,342, 7,692,696, 7,599,577, 7,606,417, 7,747,596, 7,506,057, 7,685,341, 7,694,048, 7,715,597, 7,565,030,

7,636,486, 7,639,888, 7,536,036, 7,738,015, 7,590,305, 7,352,394, 7,564,994, 7,315,658, 7,630,006, 7,440,593, and 7,317,815, and

U.S. patent application Ser. Nos. 13/306,568, 13/282,458, 13/234,149, 13/234,146, 13/234,139, 13/220,612, 13/084,340, 13/078,971, 13/077,936, 13/077,891, 13/035,907, 13/028,203, 13/020,805, 12/959,320, 12/944,701 and 12/944,662, and

United States published patent applications serial nos. 2012/0019614, 2012/0019613, 2012/0008002, 2011/0216156, 2011/0205381, 2012/0007942, 2011/0141227, 2011/0002506, 2011/0102553, 2010/0329582, 2011/0007174, 20100321537, 2011/0141226, 2010/0141787, 2011/0081052, 2010/0066822, 2010/0026831, 2009/0303343, 2009/0238419, 2010/0272363, 2009/0189998, 2009/0189997, 2009/0190803, 2009/0179999, 2009/0167893, 2009/0179998, 2008/0309769, 2008/0266419, 2008/0220750, 2008/0219517, 2009/0196466, 2009/0123063, 2008/0112599, 2009/0080713, 2009/0080797, 2009/0080796, 2008/0219581, 2009/0115915, 2008/0309770, 2007/0296833 and 2007/0269108.

Auto-focus features may be included in a camera or camera module as described at US published patent application no. 2012/0075492 and/or U.S. application Ser. Nos. 12/944,701, 12/944,703, 13/020,805, 13/077,891 and 13/077,936.

Features described at U.S. application Ser. Nos. 13/028,203, 13/028,205 and 13/028,206 are incorporated by reference and may also be used in alternative embodiments to register images captured that have global or local rotation between them and/or to discern the motion of the camera module and/or one or more objects in a captured scene.

It will be appreciated that the illustrated embodiment is provided for exemplary purposes only and that many variations of the implementation are possible. For example, some functionality shown as being implemented in one module could be migrated to other modules.

In the illustrated embodiment, tiles have been described as rectangular and defined by four nodes. However, it will be appreciated that although more complex, the invention could also be implemented with non-rectangular tiles defined by 3 or more nodes; and indeed the local grid need not be defined by a uniform array of tiles. Instead, these could in certain applications be non-uniform.

The invention is not limited to the embodiment(s) described herein but can be amended or modified without departing from the scope of the present invention, as set forth in the appended claims and structural and functional equivalents thereof.

What is claimed is:

1. A method of correcting a distorted input image comprising:

determining a local region of an image to be displayed and dividing said local region into an array of tiles, each tile having a boundary defined by a plurality of nodes, each node having a set of coordinates within said an image space of the image;

for each tile of the local region, transforming said node coordinates according to a first local transformation to take into account a local distortion of said input image in said local region;

determining a second global transformation mapping coordinates for a global array of nodes to take into account a global distortion of said input image;

for each node of each tile of the local region:

i) determining nodes of the global array immediately surrounding the node;

13

ii) interpolating the transformation of the surrounding global node coordinates to transform the at least locally transformed coordinates of the node according to the global transformation;

for each tile of the local region:

i) reading a non-rectangular portion of said distorted input image corresponding to the locally and globally transformed node coordinates of said tile;

ii) correcting the distortion of said non-rectangular portion of said distorted input image to provide a tile of a corrected output image; and

storing said corrected tiles for said output image.

2. A method according to claim 1 wherein said tiles of said local region and said corrected tiles of said output image are rectangular and defined by four nodes.

3. A method according to claim 1 further comprising:
determining an affine transformation applicable to at least a portion of the nodes of said local region of an image;
and
transforming at least said portion of the node coordinates of said local region according to said affine transformation.

4. A method according to claim 3 wherein said affine transformation comprises multiplying input node coordinates u, v with the following matrix transformation:

$$\begin{bmatrix} u' \\ v' \end{bmatrix} = \begin{bmatrix} a1 & a2 \\ a3 & a4 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} + \begin{bmatrix} a5 \\ a6 \end{bmatrix}$$

to produce transformed coordinate values u', v' .

5. A method according to claim 4 wherein said matrix coefficients $a1 \dots a6$ are changed for successive different subsets of nodes of the local region to compensate for rolling shutter distortion.

6. A method according to claim 4 wherein said matrix coefficients $a1 \dots a6$ are changed for different subsets of nodes of the local region to apply a user defined distortion to said input image.

7. A method according to claim 4 wherein said matrix coefficients $a1 \dots a6$ are set to compensate for inter-frame camera shake.

8. A method according to claim 1 wherein said affine transformation is applied to said node coordinates either before or after said local transforming.

9. A method according to claim 1 further comprising, for each tile of the local region, determining maximum and minimum memory address locations of successive rows of said input image sufficient to span said tile boundary, said maximum and minimum addresses for each of tile the input image corresponding to a non-rectangular region of the input image.

10. A method according to claim 9 further comprising extending the memory locations to be read from each row of said distorted input image to ensure at least four pixels immediately outside the distorted tile boundary are available for correcting the distortion of said tile.

11. A method according to claim 1 comprising acquiring said input image with a wide field of view, WFOV, lens system and wherein said global transformation is arranged to compensate for non-uniformities in said WFOV lens system.

12. A method according to claim 1 comprising acquiring said input image with a zoom lens system and responsive to changing image acquisition focal length, changing said global transformation to compensate for different non-uniformities in said lens system.

14

13. A method according to claim 1 comprising changing one of said global or said local transformation in response to adjusting an angle of an image acquisition sensor relative to a lens system.

14. A method according to claim 13 wherein said angle comprises a rotation angle about an axis normal to a plane of the image sensor.

15. A method according to claim 13 wherein said angle comprises a rotation angle about an axis parallel to a plane of the image sensor.

16. A method according to claim 1 wherein said local transformation is arranged to compensate for changes in perspective caused by a lens system for different local regions of said input image.

17. A method for correcting a distorted input image comprising:

determining a local region of an image to be displayed and dividing said region into an array of rectangular tiles, each tile corresponding to a distorted tile with a non-rectangular boundary within said input image;

for each tile of the local region:

determining maximum and minimum memory address locations of successive rows of said input image sufficient to span said boundary of said distorted tile;

reading successive rows of said distorted input from between said maximum and minimum addresses; and

correcting the distortion of said non-rectangular portion of said distorted input image to provide a tile of a corrected output image; and

storing said corrected tiles for said output image.

18. A method according to claim 17 further comprising extending the memory locations to be read from each row of said distorted input image to ensure at least four pixels immediately outside the distorted tile boundary are available for correcting the distortion of said tile.

19. A method according to claim 17 further comprising extending the memory locations to be read from each row of said distorted input image sufficiently to enable interpolation of all input image pixels within and including the distorted tile boundary.

20. An image acquisition device comprising:

a lens and image sensor for acquiring a distorted input image;

a processor;

a memory for storing the distorted input image and a corrected output image;

a system bus connected to said memory; and

a distortion correction module connected to said system bus for reading distorted input image information from said memory and writing corrected output image information to said memory, said module being arranged to perform the steps of:

determining a local region of an image to be displayed and dividing said region into an array of tiles, each tile having a boundary defined by a plurality of nodes, each node having a set of coordinates within an image space of said image;

for each tile of the local region, transforming said node coordinates according to a first local transformation to take into account a local distortion of said input image in said local region;

determining a second global transformation mapping coordinates for a global array of nodes to take into account a global distortion of said input image;

for each node of each tile of the local region:

i) determining nodes of the global array immediately surrounding the node;

15

ii) interpolating the transformation of the surrounding global node coordinates to transform the at least locally transformed coordinates of the node according to the global transformation;

for each tile of the local region:

i) reading a non-rectangular portion of said distorted input image corresponding to the locally and globally transformed node coordinates of said tile;

ii) correcting the distortion of said non-rectangular portion of said distorted input image to provide a tile of a corrected output image; and

storing said corrected tiles for said output image.

21. An image acquisition device according to claim **20** wherein said tiles of said local region and said corrected tiles of said output image are rectangular and defined by four nodes.

22. An image acquisition device according to claim **20** wherein said module is further arranged to perform the steps of:

determining an affine transformation applicable to at least a portion of the nodes of said local region of an image; and

transforming at least said portion of the node coordinates of said local region according to said affine transformation.

23. An image acquisition device according to claim **22** wherein said affine transformation comprises multiplying input node coordinates u, v with the following matrix transformation:

$$\begin{bmatrix} u' \\ v' \end{bmatrix} = \begin{bmatrix} a1 & a2 \\ a3 & a4 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} + \begin{bmatrix} a5 \\ a6 \end{bmatrix}$$

to produce transformed coordinate values u', v' .

24. An image acquisition device according to claim **23** wherein said matrix coefficients $a1 \dots a6$ are changed for successive different subsets of nodes of the local region to compensate for rolling shutter distortion.

25. An image acquisition device according to claim **23** wherein said matrix coefficients $a1 \dots a6$ are changed for different subsets of nodes of the local region to apply a user defined distortion to said input image.

26. An image acquisition device according to claim **23** wherein said matrix coefficients $a1 \dots a6$ are set to compensate for inter-frame camera shake.

27. An image acquisition device according to claim **20** wherein said affine transformation is applied to said node coordinates either before or after said local transforming.

28. An image acquisition device according to claim **20** wherein said module is further arranged to perform the steps of: for each tile of the local region, determining maximum and minimum memory address locations of successive rows of said input image sufficient to span said tile boundary, said maximum and minimum addresses for each of tile the input image corresponding to a non-rectangular region of the input image.

29. An image acquisition device according to claim **28** wherein said module is further arranged to perform the steps of: extending the memory locations to be read from each row of said distorted input image to ensure at least four pixels immediately outside the distorted tile boundary are available for correcting the distortion of said tile.

30. An image acquisition device according to claim **20** comprising acquiring said input image with a wide field of

16

view, WFOV, lens system and wherein said global transformation is arranged to compensate for non-uniformities in said WFOV lens system.

31. An image acquisition device according to claim **20** wherein said module is further arranged to perform the steps of: acquiring said input image with a zoom lens system and responsive to changing image acquisition focal length, changing said global transformation to compensate for different non-uniformities in said lens system.

32. An image acquisition device according to claim **20** wherein said module is further arranged to perform the steps of: changing one of said global or said local transformation in response to adjusting an angle of an image acquisition sensor relative to a lens system.

33. An image acquisition device according to claim **32** wherein said angle comprises a rotation angle about an axis normal to a plane of the image sensor.

34. An image acquisition device according to claim **32** wherein said angle comprises a rotation angle about an axis parallel to a plane of the image sensor.

35. An image acquisition device according to claim **20** wherein said local transformation is arranged to compensate for changes in perspective caused by a lens system for different local regions of said input image.

36. One or more non-transitory processor-readable media having code embedded therein for programming a processor to perform a method of correcting a distorted input image, wherein the method comprises:

determining a local region of an image to be displayed and dividing said region into an array of tiles, each tile having a boundary defined by a plurality of nodes, each node having a set of coordinates within an image space of said image;

for each tile of the local region, transforming said node coordinates according to a first local transformation to take into account a local distortion of said input image in said local region;

determining a second global transformation mapping coordinates for a global array of nodes to take into account a global distortion of said input image;

for each node of each tile of the local region:

i) determining nodes of the global array immediately surrounding the node;

ii) interpolating the transformation of the surrounding global node coordinates to transform the at least locally transformed coordinates of the node according to the global transformation;

for each tile of the local region:

i) reading a non-rectangular portion of said distorted input image corresponding to the locally and globally transformed node coordinates of said tile;

ii) correcting the distortion of said non-rectangular portion of said distorted input image to provide a tile of a corrected output image; and

storing said corrected tiles for said output image.

37. One or more non-transitory processor-readable media according to claim **36** wherein said tiles of said local region and said corrected tiles of said output image are rectangular and defined by four nodes.

38. One or more non-transitory processor-readable media according to claim **36** wherein said method further comprises the steps of:

determining an affine transformation applicable to at least a portion of the nodes of said local region of an image; and

17

transforming at least said portion of the node coordinates of said local region according to said affine transformation.

39. One or more non-transitory processor-readable media according to claim 36 wherein said affine transformation is applied to said node coordinates either before or after said local transforming.

40. One or more non-transitory processor-readable media according to claim 38 wherein said affine transformation comprises multiplying input node coordinates u, v with the following matrix transformation:

$$\begin{bmatrix} u' \\ v' \end{bmatrix} = \begin{bmatrix} a1 & a2 \\ a3 & a4 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} + \begin{bmatrix} a5 \\ a6 \end{bmatrix}$$

to produce transformed coordinate values u', v' .

41. One or more non-transitory processor-readable media according to claim 40 wherein said matrix coefficients $a1 \dots a6$ are changed for successive different subsets of nodes of the local region to compensate for rolling shutter distortion.

42. One or more non-transitory processor-readable media according to claim 40 wherein said matrix coefficients $a1 \dots a6$ are changed for different subsets of nodes of the local region to apply a user defined distortion to said input image.

43. One or more non-transitory processor-readable media according to claim 40 wherein said matrix coefficients $a1 \dots a6$ are set to compensate for inter-frame camera shake.

44. One or more non-transitory processor-readable media according to claim 36 wherein said method further comprises the steps of: for each tile of the local region, determining maximum and minimum memory address locations of successive rows of said input image sufficient to span said tile boundary, said maximum and minimum addresses for each of tile the input image corresponding to a non-rectangular region of the input image.

45. One or more non-transitory processor-readable media according to claim 44 wherein said method further comprises the steps of: extending the memory locations to be read from each row of said distorted input image to ensure at least four pixels immediately outside the distorted tile boundary are available for correcting the distortion of said tile.

46. One or more non-transitory processor-readable media according to claim 36 comprising acquiring said input image with a wide field of view, WFOV, lens system and wherein said global transformation is arranged to compensate for non-uniformities in said WFOV lens system.

47. One or more non-transitory processor-readable media according to claim 36 wherein said method further comprises the steps of: acquiring said input image with a zoom lens system and responsive to changing image acquisition focal length, changing said global transformation to compensate for different non-uniformities in said lens system.

48. One or more non-transitory processor-readable media according to claim 36 wherein said method further comprises the steps of: changing one of said global or said local transformation in response to adjusting an angle of an image acquisition sensor relative to a lens system.

49. One or more non-transitory processor-readable media according to claim 48 wherein said angle comprises a rotation angle about an axis normal to a plane of the image sensor.

50. One or more non-transitory processor-readable media according to claim 48 wherein said angle comprises a rotation angle about an axis parallel to a plane of the image sensor.

51. One or more non-transitory processor-readable media according to claim 36 wherein said local transformation is

18

arranged to compensate for changes in perspective caused by a lens system for different local regions of said input image.

52. An image acquisition device comprising:
a lens and image sensor for acquiring a distorted input image;

a processor;

a memory for storing the distorted input image and a corrected output image;

a system bus connected to said memory; and

a distortion correction module connected to said system bus for reading distorted input image information from said memory and writing corrected output image information to said memory, said module being arranged to perform the steps of:

determining a local region of an image to be displayed and dividing said region into an array of rectangular tiles, each tile corresponding to a distorted tile with a non-rectangular boundary within said input image;

for each tile of the local region:

determining maximum and minimum memory address locations of successive rows of said input image sufficient to span said boundary of said distorted tile;

reading successive rows of said distorted input from between said maximum and minimum addresses; and

correcting the distortion of said non-rectangular portion of said distorted input image to provide a tile of a corrected output image; and

storing said corrected tiles for said output image.

53. An image acquisition device according to claim 52 wherein said module is further arranged to perform the steps of: extending the memory locations to be read from each row of said distorted input image to ensure at least four pixels immediately outside the distorted tile boundary are available for correcting the distortion of said tile.

54. An image acquisition device according to claim 52 wherein said module is further arranged to perform the steps of: extending the memory locations to be read from each row of said distorted input image sufficiently to enable interpolation of all input image pixels within and including the distorted tile boundary.

55. One or more non-transitory processor-readable media having code embedded therein for programming a processor to perform a method of correcting a distorted input image, wherein the method comprises:

determining a local region of an image to be displayed and dividing said region into an array of rectangular tiles, each tile corresponding to a distorted tile with a non-rectangular boundary within said input image;

for each tile of the local region:

determining maximum and minimum memory address locations of successive rows of said input image sufficient to span said boundary of said distorted tile;

reading successive rows of said distorted input from between said maximum and minimum addresses; and

correcting the distortion of said non-rectangular portion of said distorted input image to provide a tile of a corrected output image; and

storing said corrected tiles for said output image.

56. One or more non-transitory processor-readable media according to claim 55 wherein said method further comprises the steps of: extending the memory locations to be read from each row of said distorted input image to ensure at least four pixels immediately outside the distorted tile boundary are available for correcting the distortion of said tile.

57. One or more non-transitory processor-readable media according to claim 55 wherein said method further comprises the steps of: extending the memory locations to be read from

each row of said distorted input image sufficiently to enable interpolation of all input image pixels within and including the distorted tile boundary.

* * * * *